floppy disk drive, a magnetic tape drive, an optical disk drive, a flash drive, or the like. Storage interface **134** generally represents any type or form of interface or device for transferring data between storage devices **132** and **133** and other components of computing system **110**.

[0041] In one example, databases **140** may be stored in primary storage device **132**. Databases **140** may represent portions of a single database or computing device or it may represent multiple databases or computing devices. For example, databases **140** may represent (be stored on) a portion of computing system **110**. Alternatively, databases **140** may represent (be stored on) one or more physically separate devices capable of being accessed by a computing device, such as computing system **110**.

[0042] Continuing with reference to FIG. **1**, storage devices **132** and **133** may be configured to read from and/or write to a removable storage unit configured to store computer software, data, or other computer-readable information. Examples of suitable removable storage units include, without limitation, a floppy disk, a magnetic tape, an optical disk, a flash memory device, or the like. Storage devices **132** and **133** may also include other similar structures or devices for allowing computer software, data, or other computer-readable instructions to be loaded into computing system **110**. For example, storage devices **132** and **133** may be configured to read and write software, data, or other computer-readable information. Storage devices **132** and **133** may also be a part of computing system **110** or may be separate devices accessed through other interface systems.

[0043] Many other devices or subsystems may be connected to computing system **110**. Conversely, all of the components and devices illustrated in FIG. **1** need not be present to practice the embodiments described herein. The devices and subsystems referenced above may also be interconnected in different ways from that shown in FIG. **1**. Computing system **110** may also employ any number of software, firmware, and/or hardware configurations. For example, the example embodiments disclosed herein may be encoded as a computer program (also referred to as computer software, software applications, computer-readable instructions, or computer control logic) on a computer-readable medium.

[0044] The computer-readable medium containing the computer program may be loaded into computing system **110**. All or a portion of the computer program stored on the computer-readable medium may then be stored in system memory **116** and/or various portions of storage devices **132** and **133**. When executed by processor **114**, a computer program loaded into computing system **110** may cause processor **114** to perform and/or be a means for performing the functions of the example embodiments described and/or illustrated herein. Additionally or alternatively, the example embodiments described and/or illustrated herein may be implemented in firmware and/or hardware.

[0045] FIG. **2** is a block diagram of an example of a network architecture **200** in which client systems **210**, **220**, and **230** and servers **240** and **245** may be coupled to a network **250**. Client systems **210**, **220**, and **230** generally represent any type or form of computing device or system, such as computing system **110** of FIG. **1**.

[0046] Similarly, servers **240** and **245** generally represent computing devices or systems, such as application servers or database servers, configured to provide various database services and/or run certain software applications. Network **250** generally represents any telecommunication or computer network including, for example, an intranet, a wide area network (WAN), a local area network (LAN), a personal area network (PAN), or the Internet.

[0047] With reference to computing system **110** of FIG. **1**, a communication interface, such as communication interface **122**, may be used to provide connectivity between each client system **210**, **220**, and **230** and network **250**. Client systems **210**, **220**, and **230** may be able to access information on server **240** or **245** using, for example, a Web browser or other client software. Such software may allow client systems **210**, **220**, and **230** to access data hosted by server **240**, server **245**, storage devices **260**(**1**)-(L), storage devices **270**(**1**)-(N), storage devices **290**(**1**)-(M), or intelligent storage array **295**.

[0048] In one embodiment, all or a portion of one or more of the example embodiments disclosed herein are encoded as a computer program and loaded onto and executed by a switch device connected between servers **240** or **245** and client systems **210**, **220**, and **230**.

[0049] Applications of Processing Packets Which Contain Geographic Location Information of the Packet Sender

[0050] In one embodiment, the present invention provides a method and apparatus that include geo-location information in IP packets transmitted by a client device in an efficient and flexible manner. Further, in one embodiment, the present invention provides a method for authenticating packets that originate from a client device based on the geographic location of the client. Additionally, in one embodiment, the present invention provides a method for prioritizing or efficiently routing packets based on the geographic location of the client device. The geo-location information inserted in an extension header of an IPv6 packet can be used by Distributed Denial of Serve (DDoS) applications, Application Delivery Controllers (ADCs), Cloud and other monitoring applications.

[0051] In one embodiment, an extension header in an IPv6 packet is used to transmit the geo-location of the client device. FIG. **3** is an illustration of the IPv6 header format. The IPv6 header comprises a "Next Header" field **312**, which is an 8-bit selector field to identify the type of header immediately following the IPv6 header. The Next Header field **312** can be used to indicate that the extension header following the main IPv6 header comprises geo-location information.

[0052] Further, the IPv6 header comprises a source address field **313** and a destination address field **314** to indicate the source and destination address of the IP packet, wherein the source IP address of the client is indicated by field **313** and the destination IP address of the server or load balancer is indicated by field **314**.

[0053] FIG. **4** illustrates the IPv6 extension header format. As illustrated in FIG. **4**, an IPv6 packet can have zero, one or more extension headers, each identified by the Next Header field **312** of the preceding header. In one embodiment of the present invention, one of the extension headers is used to convey the geo-location information of the sender. The Next Header field **412** in a preceding extension header (or the Next Header field **312** in the main header) is used to encode a prefix code to indicate that the next header comprises geo-location information, e.g., using "GEO" as a prefix code. The extension header code will likely need to be assigned by the Internet Assigned Numbers Authority (IANA). The "GEO" prefix code will be referred to herein as an alias to this assigned code.

[0054] In one embodiment, a type of header known as a Destination Options header may be used to encode geo-loca-